Golem95:
calculating tensor integrals with up to six external legs numerically

Gudrun Heinrich

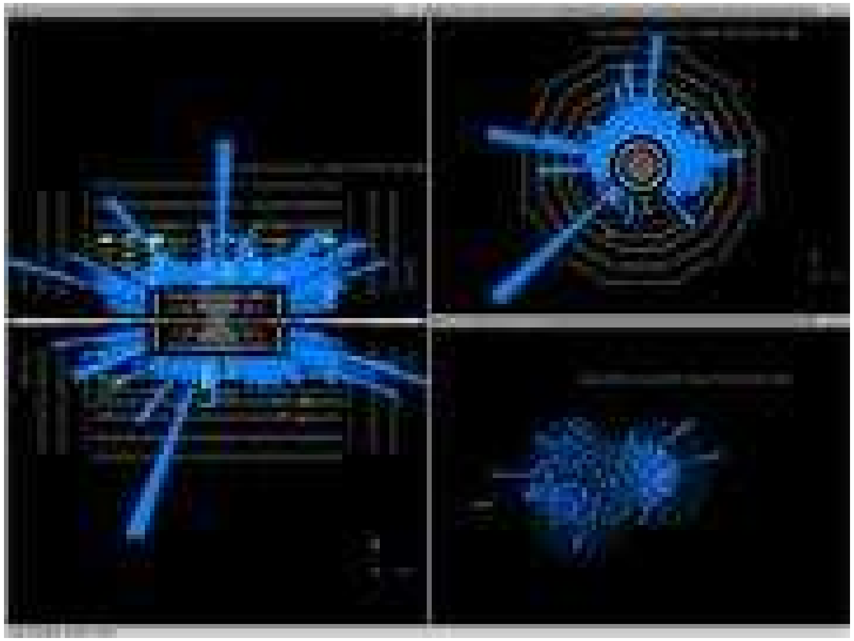
University of Durham

Institute for Particle Physics Phenomenology

in collaboration with T.Binoth, J.-Ph.Guillet, E.Pilon, T.Reiter

Exploring the TeV scale

with LHC (or the Tevatron already ?) we are entering a
New Era in Particle Physics !



The LHC

- will shed light on the origin of mass ("Higgs mechanism")
- may discover supersymmetry/extra dimensions, provide information about dark matter

The LHC

- will shed light on the origin of mass ("Higgs mechanism")
- may discover supersymmetry/extra dimensions, provide information about dark matter
- 1 Terabyte of data every day
- ~ 1000 hadronic tracks in detector per event

  proton remnants or high energy interactions between quarks/gluons (QCD)

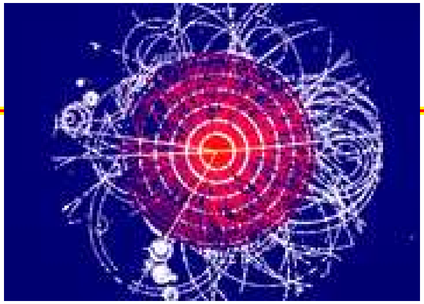
The LHC

- will shed light on the origin of mass ("Higgs mechanism")
- may discover supersymmetry/extra dimensions, provide information about dark matter
- 1 Terabyte of data every day
- ~ 1000 hadronic tracks in detector per event
  proton remnants or high energy interactions between quarks/gluons (QCD)

process	events/sec	
QCD jets $E_T > 150\,$GeV	100	background
$W \to e\nu$	15	background
$t\bar{t}$	1	background
Higgs, $m_H \sim 130\,$GeV	0.02	signal
gluinos, $m \sim 1\,$TeV	0.001	signal

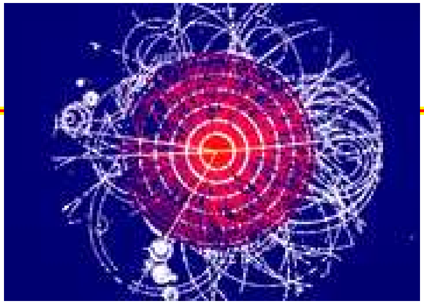
$\Rightarrow$ enormous backgrounds !

preparing for the LHC



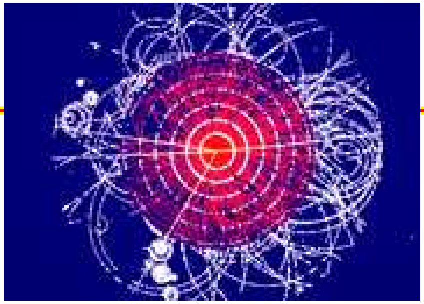
- we might see very clear signatures
  e.g. 4 highly energetic leptons

preparing for the LHC



- we might see very clear signatures
  e.g. 4 highly energetic leptons
- we might have to work hard to claim a discovery
  e.g. light Higgs boson $m_H \sim 120\,$GeV

preparing for the LHC

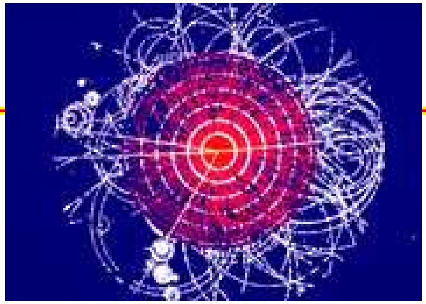


- we might see very clear signatures
  e.g. 4 highly energetic leptons
- we might have to work hard to claim a discovery
  e.g. light Higgs boson $m_H \sim 120\,$GeV

  $\Rightarrow$ be prepared !
- we first have to "rediscover" the Standard Model, control jet energy scale, underlying event, ...
- maximal control of theory expectations

  for signals <u>and</u> backgrounds is required
- measuring the backgrounds is not always possible
  e.g. neutrinos in final state

preparing for the LHC



- we might see very clear signatures
  e.g. 4 highly energetic leptons
- we might have to work hard to claim a discovery
  e.g. light Higgs boson $m_H \sim 120\,$GeV

  $\Rightarrow$ be prepared !
- we first have to "rediscover" the Standard Model, control jet energy scale, underlying event, . . .
- maximal control of theory expectations
  for signals and backgrounds is required
- measuring the backgrounds is not always possible
  e.g. neutrinos in final state
- need to have **precise** theory predictions

HP2

need High Precision for Hard Processes !

HP2

need High Precision for Hard Processes !

predictions based on Leading Order (LO) in perturbation theory are not sufficient

HP2

need High Precision for Hard Processes !

predictions based on Leading Order (LO) in perturbation theory are not sufficient

LHC: many interesting processes lead to multi-particle ($2 \rightarrow 3, 4, \ldots$) final states, e.g.

$pp \rightarrow q\,q\,H \rightarrow W^+W^- + 2$ jets, $pp \rightarrow H + t\bar{t} \rightarrow b\bar{b}\,t\bar{t}$, ...

HP2

need High Precision for Hard Processes !

predictions based on Leading Order (LO) in perturbation theory are not sufficient

LHC: many interesting processes lead to multi-particle ($2 \to 3, 4, \ldots$) final states, e.g.

$pp \to q\, q\, H \to W^+ W^- + 2$ jets, $pp \to H + t\bar{t} \to b\bar{b}\, t\bar{t}, \ldots$

to calculate them at NLO : HT2 (Hard Thinking for HEP Theorists)

HP2

need High Precision for Hard Processes !

predictions based on Leading Order (LO) in perturbation theory are not sufficient

$\boxed{\text{LHC:}}$ many interesting processes lead to multi-particle ($2 \to 3, 4, \ldots$) final states, e.g.

$pp \to q\,q\,H \to W^+W^- + 2$ jets, $pp \to H + t\bar{t} \to b\bar{b}\,t\bar{t}, \ldots$

to calculate them at NLO : HT2 (Hard Thinking for HEP Theorists)

$\Rightarrow$ better methods, more automation

Automation

lots of progress recently !

- automated subtraction for NLO real radiation
  Frederix/Gehrmann/Greiner, Hasegawa/Moch/Uwer, Tevlin/Seymour, Gleisberg/Krauss 07-08

Automation

lots of progress recently !

- automated subtraction for NLO real radiation
  Frederix/Gehrmann/Greiner, Hasegawa/Moch/Uwer, Tevlin/Seymour, Gleisberg/Krauss 07-08
- new tools based on numerical implementation of unitarity cuts
  Ossola/Papadopoulos/Pittau (CutTools),
  Berger/Bern/Dixon/Febres-Cordero/Forde/Ita/Kosower/Maître (BlackHat),
  Giele/Zanderighi (Rocket),
  Ellis/Giele/Kunszt/Melnikov 07-08

Automation

lots of progress recently !

- automated subtraction for NLO real radiation
  Frederix/Gehrmann/Greiner, Hasegawa/Moch/Uwer, Tevlin/Seymour, Gleisberg/Krauss 07-08
- new tools based on numerical implementation of unitarity cuts
  Ossola/Papadopoulos/Pittau (CutTools),
  Berger/Bern/Dixon/Febres-Cordero/Forde/Ita/Kosower/Maître (BlackHat),
  Giele/Zanderighi (Rocket),
  Ellis/Giele/Kunszt/Melnikov 07-08
- new developments within methods based on Feynman diagrams
  Bredenstein/Denner/Dittmaier/Pozzorini,
  Hahn/Ilana/Rauch (FeynArts/FormCalc/LoopTools),
  Golem, Passarino et al., Yuasa et al. (GraceNLO), . . .

Methods for one-loop amplitudes

- algebraic reduction
  (pioneered by Passarino/Veltman)
  generates factorial growth in complexity

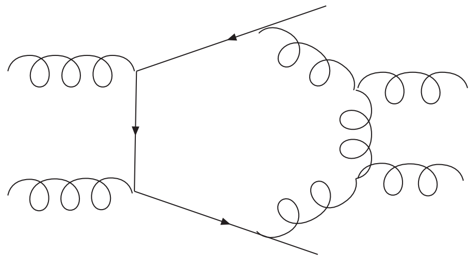
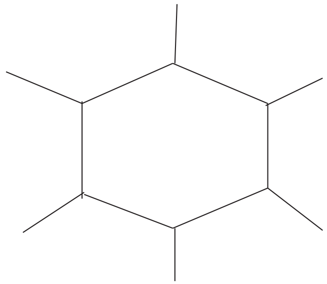
Methods for one-loop amplitudes

- algebraic reduction
  (pioneered by Passarino/Veltman)
  generates factorial growth in complexity
- fully numerical
  (pioneered by D.Soper)
  needs extraction of poles beforehand

Methods for one-loop amplitudes

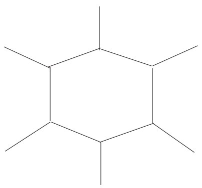
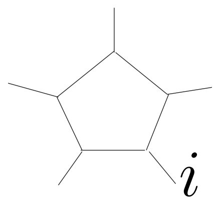
- algebraic reduction
  (pioneered by Passarino/Veltman)
  generates factorial growth in complexity
- fully numerical
  (pioneered by D.Soper)
  needs extraction of poles beforehand
- unitarity-based ("string/twistor inspired")
  (pioneered by Bern, Dixon, Dunbar, Kosower '94,
  Britto, Cachazo, Feng, Witten '04,
  Ossola, Papadopoulos, Pittau '06)
  needs special treatment of rational parts

algebraic reduction

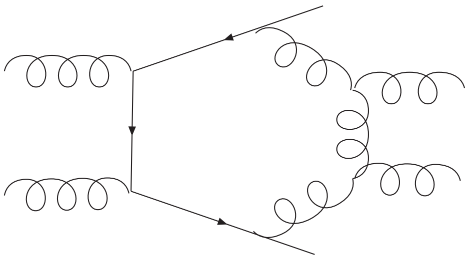
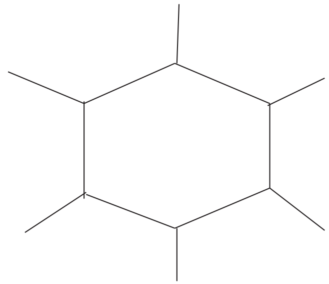
 $\rightarrow$  + integrals with less legs

non-trivial tensor structure

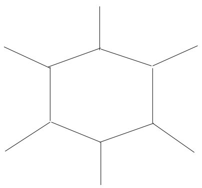
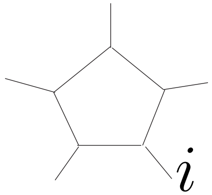
scalar 6-point function

 $= \sum_{i=1}^{6} b_i$  i . . . factorial growth in complexity !

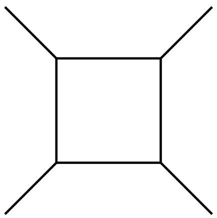
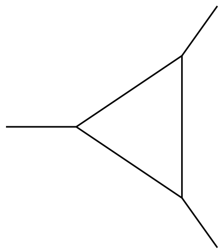
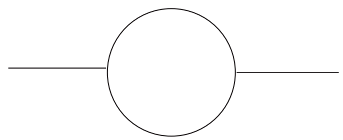
algebraic reduction

 $\longrightarrow$  + integrals with less legs

non-trivial tensor structure — scalar 6-point function

 $= \sum_{i=1}^{6} b_i$  i . . . factorial growth in complexity !

reduction to set of basis integrals (4-, 3- and 2-point funcs.)

$\mathcal{A} = C_4$  $+\ C_3$  $+\ C_2$  $+\ \mathcal{R}$

reduction to scalar basis integrals

main problems with reduction based on Feynman diagrams:

- sheer complexity of the expressions

  $\Rightarrow$ slow programs

reduction to scalar basis integrals

main problems with reduction based on Feynman diagrams:

- sheer complexity of the expressions

  $\Rightarrow$ slow programs

- reduction coefficients C_i contain inverse determinants of kinematic variables

  ("Gram determinants" detG)

  if detG $\to 0$ in certain phase space regions

  $\Rightarrow$ numerical problems

Golem approach

one possible solution: semi-numerical method

[Binoth, Guillet, GH 00], [Binoth, GH, Kauer 03], [Binoth, Guillet, GH, Pilon, Schubert 05]

combine virtues of numerical and algebraic methods

Golem approach

one possible solution: semi-numerical method

[Binoth, Guillet, GH 00], [Binoth, GH, Kauer 03], [Binoth, Guillet, GH, Pilon, Schubert 05]

combine virtues of numerical and algebraic methods

- do tensor reduction numerically
- reduce to scalar integrals and use analytic expressions where inverse determinants are harmless $\Rightarrow$ fast
- switch to numerical evaluation of boxes, triangles otherwise

Golem approach

one possible solution: semi-numerical method

[Binoth, Guillet, GH 00], [Binoth, GH, Kauer 03], [Binoth, Guillet, GH, Pilon, Schubert 05]

combine virtues of numerical and algebraic methods

- do tensor reduction numerically
- reduce to scalar integrals and use analytic expressions where inverse determinants are harmless $\Rightarrow$ fast
- switch to numerical evaluation of boxes, triangles otherwise
- formalism valid for massive and massless particles, arbitrary number of legs

Golem approach

one possible solution: semi-numerical method

[Binoth, Guillet, GH 00], [Binoth, GH, Kauer 03], [Binoth, Guillet, GH, Pilon, Schubert 05]

combine virtues of numerical and algebraic methods

- do tensor reduction numerically
- reduce to scalar integrals and use analytic expressions where inverse determinants are harmless $\Rightarrow$ fast
- switch to numerical evaluation of boxes, triangles otherwise
- formalism valid for massive and massless particles, arbitrary number of legs
- rational parts $\mathcal{R}$ are for free!
  complexity of expressions greatly reduced
  if $\mathcal{R}$ is projected out

form factor representation

$$I_N^{n,\,\mu_1\ldots\mu_r}(S) =$$

$$\sum_{l_1\cdots l_r\in S} \; p_{l_1}^{\mu_1}\cdots p_{l_r}^{\mu_r}\, A^{N,r}_{l_1\ldots,l_r}(S)$$

$$+ \sum_{l_1\cdots l_{r-2}\in S} \left[g^{\cdot\cdot} p^{\cdot}_{l_1}\cdots p^{\cdot}_{l_{r-2}} \right]^{\{\mu_1\cdots\mu_r\}} B^{N,r}_{l_1\ldots,l_{r-2}}(S)$$

$$+ \sum_{l_1\cdots l_{r-4}\in S} \left[g^{\cdot\cdot} g^{\cdot\cdot} p^{\cdot}_{l_1}\cdots p^{\cdot}_{l_{r-4}} \right]^{\{\mu_1\cdots\mu_r\}} C^{N,r}_{l_1\ldots,l_{r-4}}(S)$$

form factor representation

$$I_N^{n,\mu_1\ldots\mu_r}(S) =$$

$$\sum_{l_1\cdots l_r\in S} p_{l_1}^{\mu_1}\cdots p_{l_r}^{\mu_r}\, A^{N,r}_{l_1\ldots,l_r}(S)$$

$$+ \sum_{l_1\cdots l_{r-2}\in S} \left[g^{\cdot\cdot} p^{\cdot}_{l_1}\cdots p^{\cdot}_{l_{r-2}}\right]^{\{\mu_1\cdots\mu_r\}} B^{N,r}_{l_1\ldots,l_{r-2}}(S)$$

$$+ \sum_{l_1\cdots l_{r-4}\in S} \left[g^{\cdot\cdot} g^{\cdot\cdot} p^{\cdot}_{l_1}\cdots p^{\cdot}_{l_{r-4}}\right]^{\{\mu_1\cdots\mu_r\}} C^{N,r}_{l_1\ldots,l_{r-4}}(S)$$

important: more than two metric tensors $g^{\mu\nu}$ never occur !

reason: for $N \geq 6$: simultaneous reduction of rank r and number of legs N

$$I_N^{n,\mu_1\ldots\mu_r}(S) = -\sum_{j\in S} \mathcal{C}^{\mu_1}_{j6}\, I_{N-1}^{n,\mu_2\ldots\mu_r}(S\setminus\{j\})$$

$$\mathcal{S}_{ij} = (r_i - r_j)^2 - m_i^2 - m_j^2$$

reduction algorithm schematically

diagram generation (e.g. QGRAF, FeynArts)

$$\Downarrow$$

$$A = \sum_i C_i^{\mu_1 \ldots \mu_r} \, I_{\mu_1 \ldots \mu_r}$$

$$\Downarrow$$

$$A = \sum_{\{l\}} f_l(p_i \cdot p_j, p_i \cdot \epsilon_j, \epsilon_i \cdot \epsilon_j) \, \{A^{N,r}_{\{l\}}, B^{N,r}_{\{l\}}, C^{N,r}_{\{l\}}\}$$

(Lorentz invariants $\times$ form factors)

$$\Downarrow$$

golem95

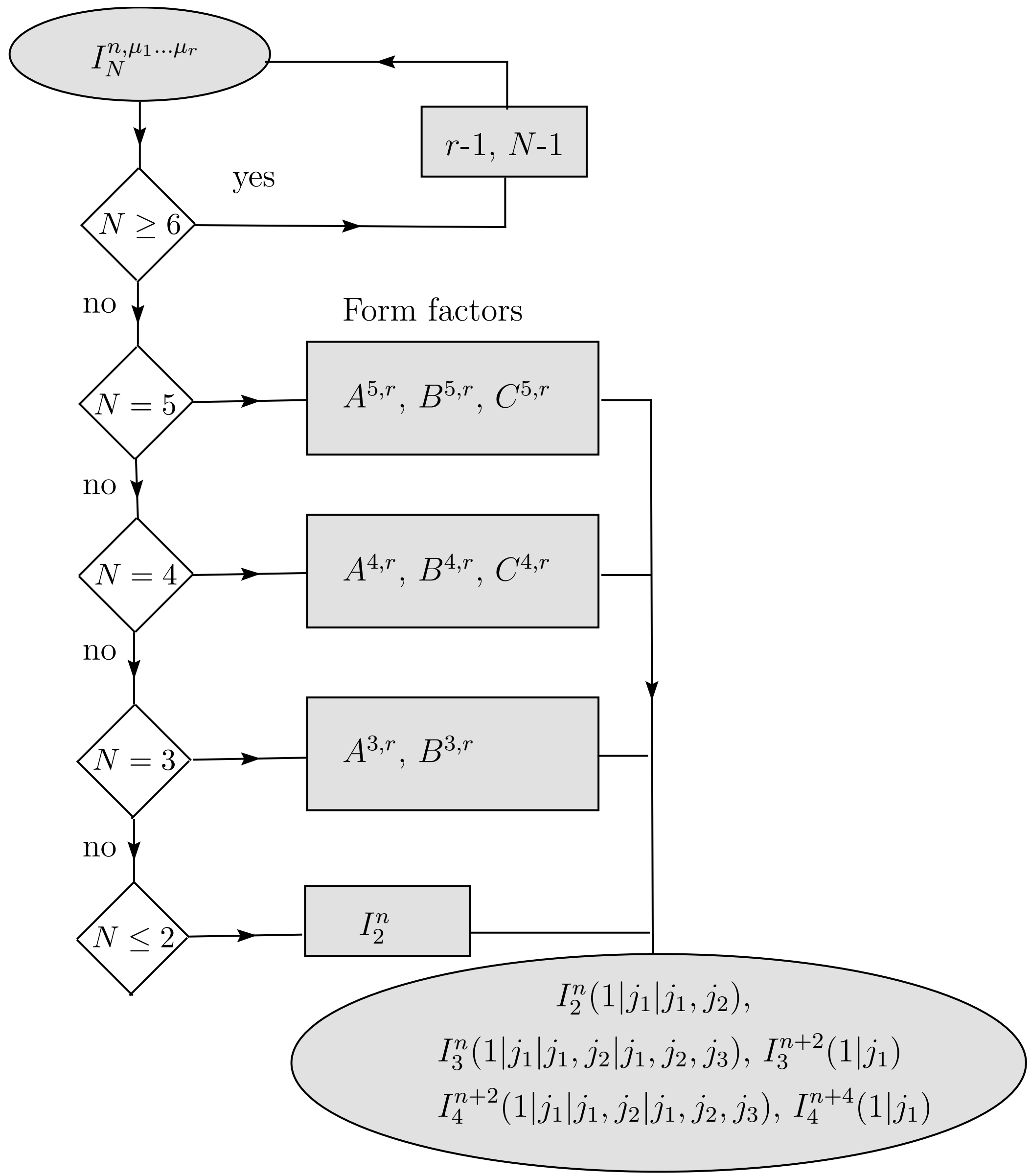
$\swarrow$ $\searrow$

numerical evaluation | reduction to scalar integrals

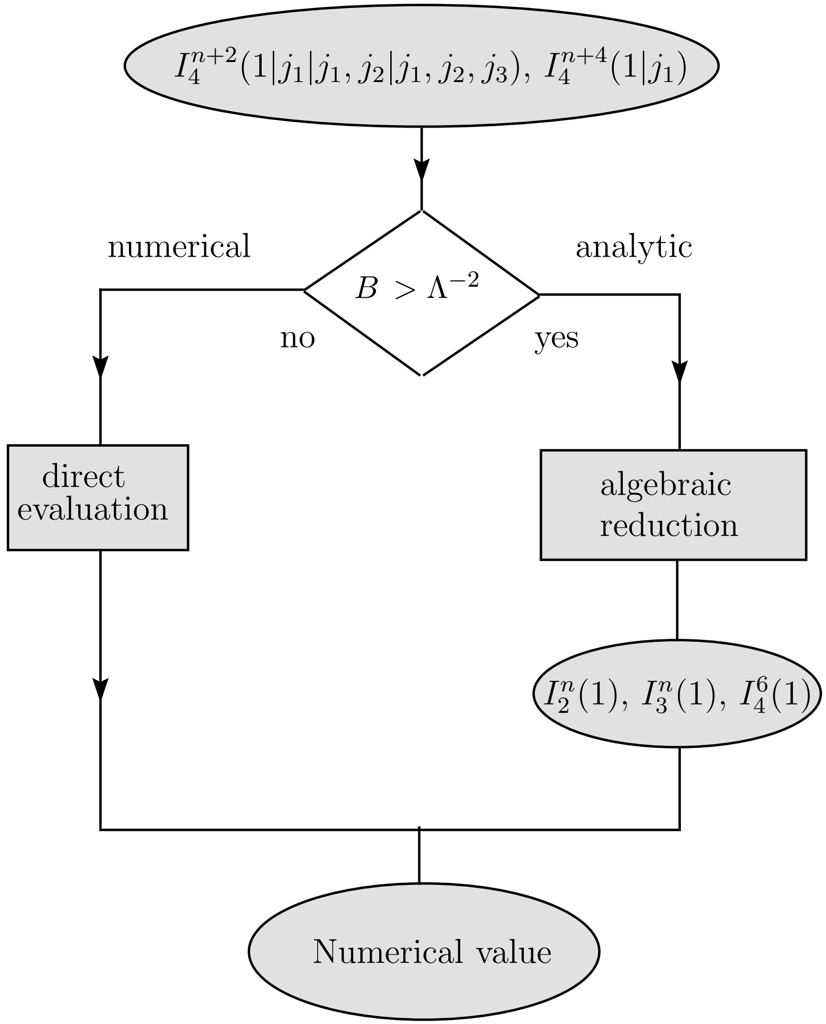
$\searrow$ $\swarrow$

numbers (Laurent series in ϵ)

from tensor integrals to parameter integrals



treatment of basis integrals



new: numerical integration based on one-dimensional parameter representation $\Rightarrow$ fast and precise

The GOLEM project and the golem95 program

golem95 code:

- calculates form factors for tensor integrals numerically
- master integrals valid for all kinematic regions, but only massless internal particles so far
- no restriction on masses of external particles
- box with all 4 legs off-shell: no one-dimensional integral representation so far $\Rightarrow$ will always be reduced to scalar box

The GOLEM project and the golem95 program

golem95 code:

- calculates form factors for tensor integrals numerically
- master integrals valid for all kinematic regions,
  but only massless internal particles so far
- no restriction on masses of external particles
- box with all 4 legs off-shell: no one-dimensional integral representation so far $\Rightarrow$ will always be reduced to scalar box

Golem project:

- include automated diagram generation,
  combine with real radiation, produce cross sections
  see T.Binoth's talk
- combine with parton shower

golem95: installation and structure

installation:

download from
http://lappweb.in2p3.fr/lapth/Golem/golem95.html
and unpack

```
./configure.pl
[–install_path=mypath] [–compiler=mycompiler]
make
make install
```

golem95: installation and structure

installation:

download from
http://lappweb.in2p3.fr/lapth/Golem/golem95.html
and unpack

./configure.pl
[–install_path=mypath] [–compiler=mycompiler]
make
make install

golem95 subdirectories:

- src: source files
- doc: documentation
- demos: 8 demo programs
- test: user interface for tests etc.

demo programs

typing configure.pl produces:

Choose which demo program you want to run:

1. three-point functions
2. four-point functions
3. five-point functions
4. six-point functions
5. calculation of 4-photon helicity amplitudes
6. numerical stability demo: $\det G \to 0$
7. numerical stability demo: $\det S \to 0$
8. Golem $\leftrightarrow$ LoopTools conventions

demo 3: rank 5 five-point

choosing option 3 will produce the following output:

you have chosen option 3: five-point functions

The Makefile has been created

Please run:

make

./comp.exe

running comp.exe will prompt for the rank:

Choose what the program should compute:

0) form factor for five-point function, rank 0

1) form factor for five-point function, rank 3 $(z_1\, z_2\, z_4)$

2) form factor for five-point function, rank 5 $(z_1\, z_2\, z_3\, z_4\, z_5)$

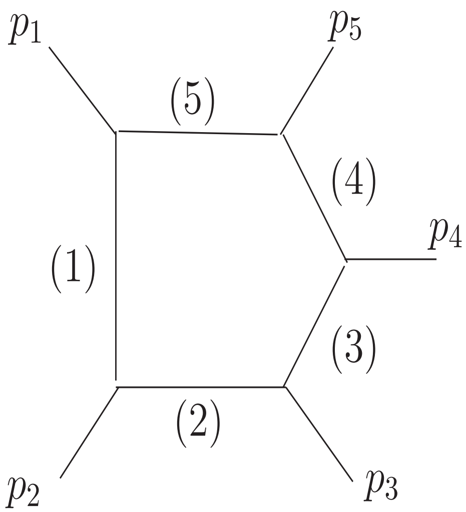
3) form factor for diagram with propagator 3 pinched, rank 0

4) form factor for diagram with propagators 1 and 4 pinched, rank 0

choosing option 2 will produce the result in about 8×10^{-3} seconds

the result written to test5point.txt looks as follows:

demo 3: rank 5 five-point



$$S(1,3) = (p_2 + p_3)^2 = -3.$$
$$S(2,4) = (p_3 + p_4)^2 = 6.$$
$$S(2,5) = (p_1 + p_2)^2 = 15.$$
$$S(3,5) = (p_4 + p_5)^2 = 2.$$
$$S(1,4) = (p_1 + p_5)^2 = -4.$$
$$S(1,2) = p_2^2 = 0.$$
$$S(2,3) = p_3^2 = 0.$$
$$S(3,4) = p_4^2 = 0.$$
$$S(4,5) = p_5^2 = 0.$$
$$S(1,5) = p_1^2 = 0.$$

A factor $\Gamma(1+\epsilon)\Gamma(1-\epsilon)^2/\Gamma(1-2\epsilon)\,(4\pi\,\mu^2)^\epsilon$ is factored out from the result.

result= $1/\epsilon^2$ * (0.0000000000E+00 + I* 0.0000000000E+00)

+ $1/\epsilon$ * (0.0000000000E+00 + I* 0.0000000000E+00)

+ (–.8615520644E-04 + I* 0.1230709464E-03)

CPU time= 7.999000000000001E-003

demo 6: Gram determinants

- reduction $N \geq 5 \to N = 4$: inverse Gram determinants completely absent
- reduction of $N \leq 4$ tensor integrals: introduces spurious 1/det(G)

$$
\begin{aligned}
I_4^{n+2}(j_1;S) &= \frac{1}{B}\left\{ b_{j_1}\, I_4^{n+2}(S) + \frac{1}{2}\sum_{j_2\in S} \mathcal{S}^{-1}_{j_1 j_2}\, I_3^n(S\setminus\{j_2\}) \right. \\
&\qquad \left. -\frac{1}{2}\sum_{j_2\in S\setminus\{j_1\}} b_{j_2}\, I_3^n(j_1;S\setminus\{j_2\}) \right\} \\
I_4^{n+2}(j_1,j_2;S) &\sim \frac{1}{B^2}\ ,\ I_4^{n+2}(j_1,j_2,j_3;S) \sim \frac{1}{B^3}\ \ldots \\
B &= \det(G)/\det(\mathcal{S})\,(-1)^{N+1} \\
\mathcal{S}_{ij} &= (r_i - r_j)^2 - m_i^2 - m_j^2 \quad ; \quad G_{ij} = 2\, r_i \cdot r_j
\end{aligned}
$$

Gram determinants

to avoid spurious 1/det(G) terms: do not reduce

golem95:

define dimensionless quantity $\hat{B} = B \times$ (largest entry of S)

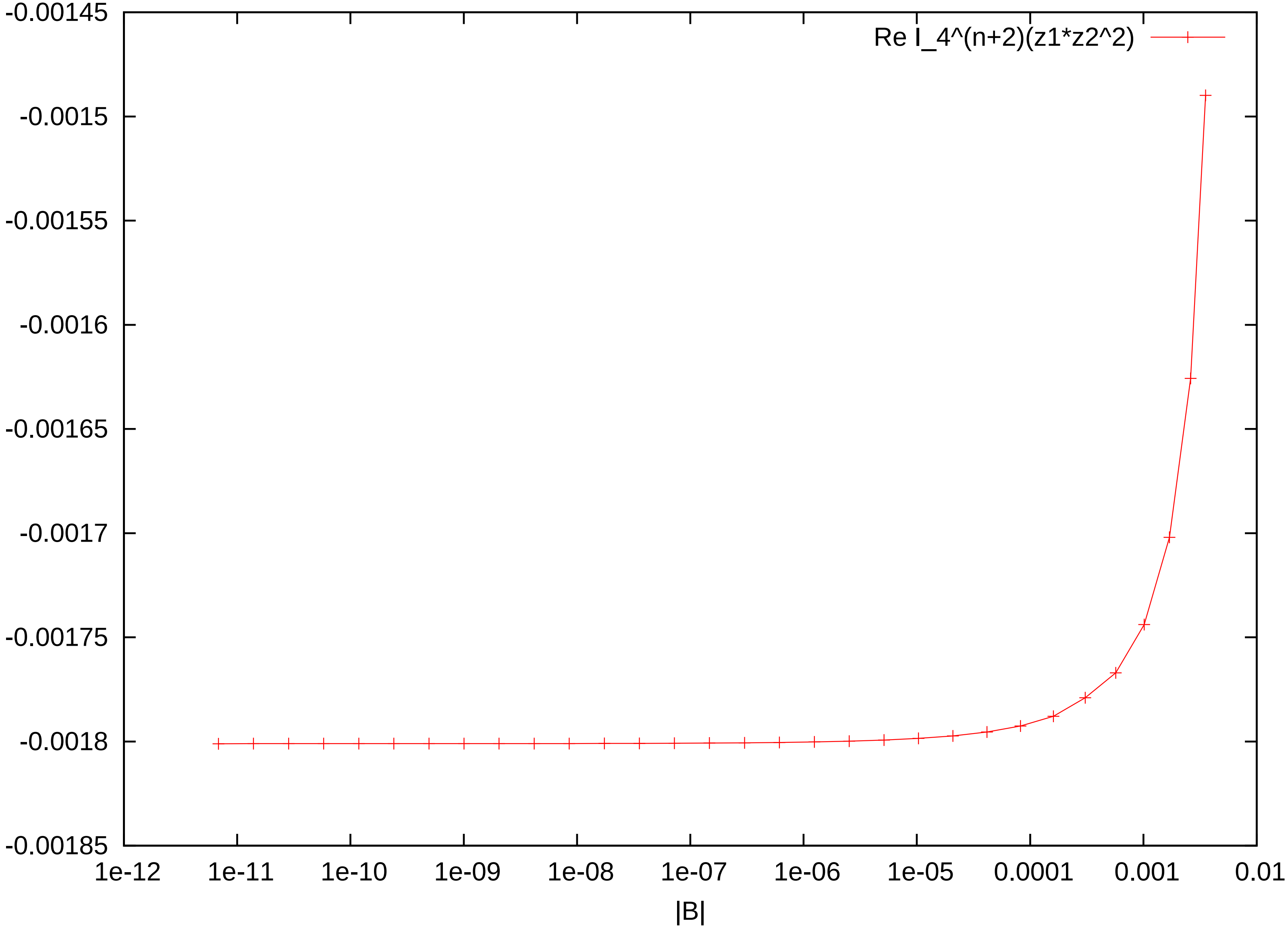
if $\hat{B} < \hat{B}^{\mathrm{cut}}$: switch to direct numerical evaluation

(default: $\hat{B}^{\mathrm{cut}} = 0.005$)

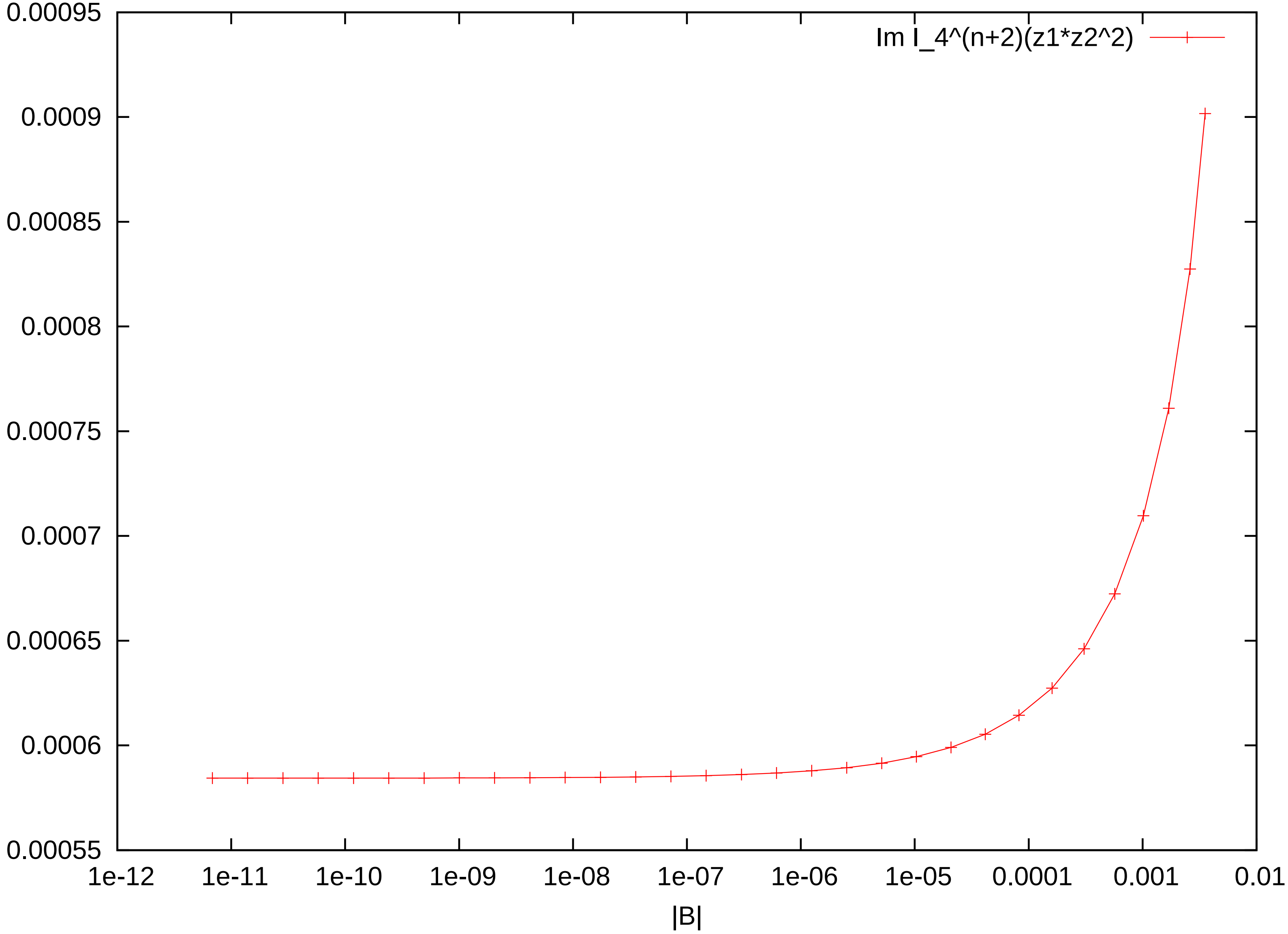
file demo_detg.f90 contains example where $\hat{B} \to 0$

in rank 3 box integral $I_4^{n+2}(1,2,2;S)$ with two massive legs

Real part for B $\rightarrow$ 0



Imaginary part for B $\rightarrow$ 0



demo 8: comparison to LoopTools

if all external legs are off-shell: master integrals IR finite
$\Rightarrow$ direct comparison to LoopTools possible

box integrals:

demo 8: comparison to LoopTools

if all external legs are off-shell: master integrals IR finite
$\Rightarrow$ direct comparison to LoopTools possible

box integrals:

pentagon integrals:

note: for $N \geq 5$ metric $g^{\mu\nu}$ can be expressed by external vectors, so definition of A_5, B_5, C_5 not unique anymore

$\Rightarrow$ comparison of contracted tensor integrals

rather than individual form factors

user-defined tests

if you would like to

- calculate certain selected numerators of a tensor form factor, or
- calculate all different numerators of a tensor form factor
- define the numerical point to be calculated
  - go to subdirectory test
  - edit the file param.input
  - define the numerical point in file momenta.dat
  - type perl maketest.pl

example: all possible form factors for rank two 6-point

user-defined tests

numerical point: ($p_i = (E_i, x_i, y_i, z_i)$):

$$
\begin{aligned}
p_1 &= (0.5, 0., 0., 0.5) \\
p_2 &= (0.5, 0., 0., -0.5) \\
p_3 &= (-0.19178191, -0.12741180, -0.08262477, -0.11713105) \\
p_4 &= (-0.33662712, 0.06648281, 0.31893785, 0.08471424) \\
p_5 &= (-0.21604814, 0.20363139, -0.04415762, -0.05710657) \\
p_6 &= (-0.2555428, -0.14270241, -0.19215546, 0.08952338)
\end{aligned}
$$

input parameters

- number of legs (only 3,4,5,6 are possible): 6
- rank: 2
- type of form factor: A, B or C
  (note: type B exists only for rank $\geq$ 2, type C exists only for rank $\geq$ 4): A
- labels of Feynman parameters in the numerator (separated by commas):
  example: put 2,2,3 for a rank 3 integral with $z_2^2\, z_3$ in the numerator
  put "all" if you want to calculate all possible numerators
  all
- name of the file containing the momenta for the numerical point to be calculated: momenta.dat
- label to distinguish different numerical points: 1

Summary

- we are on our way towards the automation of NLO calculations for multi-particle processes

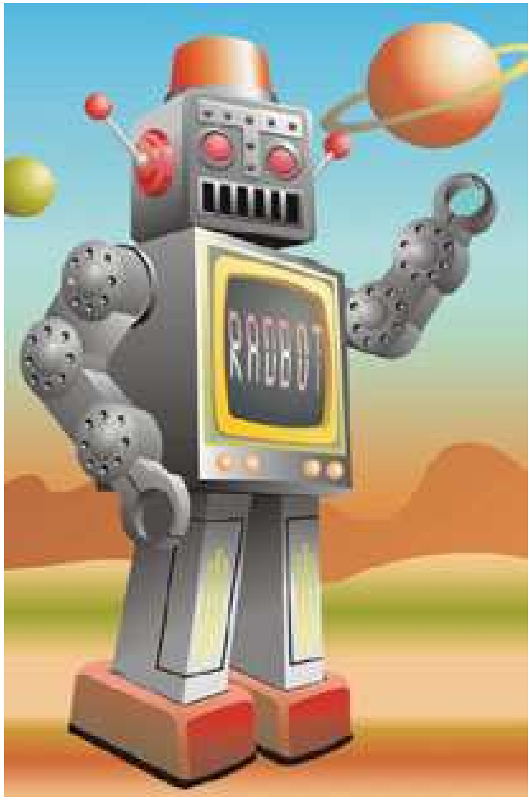
Summary

- we are on our way towards the automation of NLO calculations for multi-particle processes
- golem95 is a nice tool to have in the toolbox
  - numerically robust due to convenient basis integrals
  - can also be used as a library for master integrals
  - contains switch to compilation in quadruple precision
  - flag to calculate rational parts only
  - available at http://lappweb.in2p3.fr/lapth/Golem/golem95.html

Summary

- we are on our way towards the automation of NLO calculations for multi-particle processes
- golem95 is a nice tool to have in the toolbox
  - numerically robust due to convenient basis integrals
  - can also be used as a library for master integrals
  - contains switch to compilation in quadruple precision
  - flag to calculate rational parts only
  - available at http://lappweb.in2p3.fr/lapth/Golem/golem95.html
- ToDo:
  - add basis integrals for internal masses
  - public automated interface to amplitude generation
  - combine with automated treatment of real radiation

Golem will evolve more and more towards automation !



. . . but be careful: Stanislaw Lem's Golem XIV was so advanced that he refused to interact with those "stupid humans" . . .

backup slides

demo 7: scattering singularity

$\det \mathcal{S} \sim (\det\, G)^2 \to 0$

pentagon with $s_5 \neq 0$, else $s_j = 0$:

$$\det \mathcal{S} \;=\; 2\, s_{12}\, s_{23}\, s_{34} \left(s_{15}\, s_{45} - s_5\, s_{23}\right)$$

box (1,23,4,5):

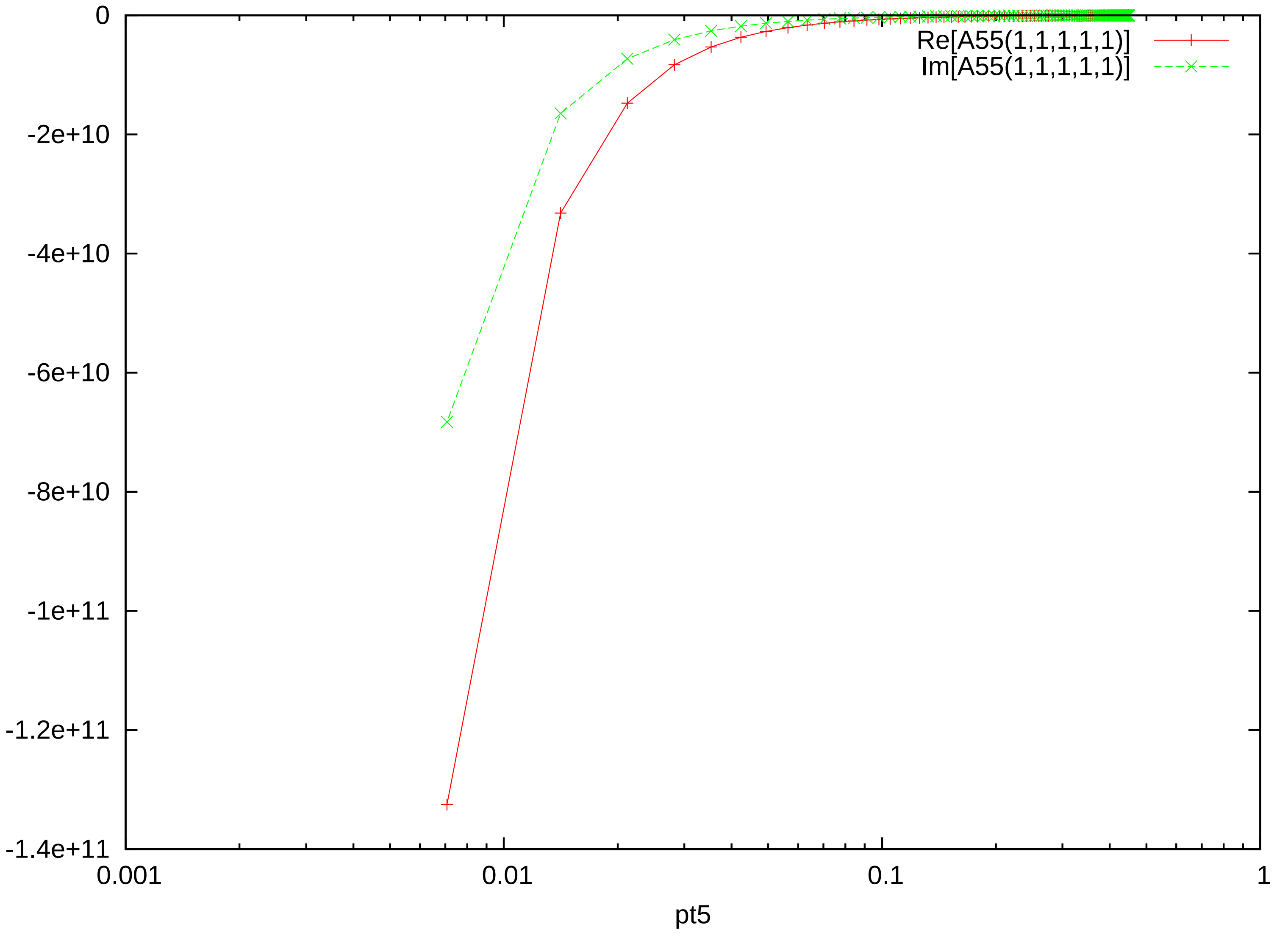
$$\det\, G \;=\; 2\, s_{14} \left(s_{15}\, s_{45} - s_5\, s_{23}\right)$$

using momentum parametrisation for 1+4 $\to$ 2+3+5

$$\det \mathcal{S} \;=\; 2\, s_{12}\, s_{23}\, s_{34}\, s_{14}\, p_{T,5}^2\;,\;\; \det G = 2\, s_{14}^2\, p_{T,5}^2$$

$p_{T,5}$: transverse momentum of particle 5 or the system 34 relative to the beam axis (z-axis)
rotation of 2,3,5 around the z-axis is evaluated to check for stability in the limit $p_{T,5} \to 0$

scattering singularity



N(N)LO wishlist for LHC (Les Houches 07)

process ($V \in \{Z, W, \gamma\}$)	relevant for	
1. $pp \to Z\,Z$ jet	$t\bar{t}H$, new physics	done
2. $pp \to t\bar{t}\,b\bar{b}$	$t\bar{t}H$	in progress
3. $pp \to t\bar{t} + 2$ jets	$t\bar{t}H$	
4. $pp \to W\,W\,W$	SUSY trilepton	done
5. $pp \to V\,V\,b\bar{b}$	VBF, new physics	
6. $pp \to V\,V + 2$ jets	VBF	
7. $pp \to V + 3$ jets	new physics	in progress
8. $pp \to b\bar{b}b\bar{b}$	H, SUSY searches	in progress
10. $\mathcal{O}(\alpha^2\alpha_s^3)\ gg \to WW$	EW sector	in progress
11. NNLO for $t\bar{t}$	benchmark, H coupl.	in progress
12. NNLO to VBF, Z/γ+j	H coupl., benchmark	in progress